



Pekka Lehtola

OULU UAS FOR STUDENTS -MOBIILISOVELLUS ANDROID-ALUSTALLE

OULU UAS FOR STUDENTS -MOBIILISOVELLUS ANDROID-ALUSTALLE

Pekka Lehtola
Opinnäytetyö
Syksy 2012
Tietotekniikan koulutusohjelma
Oulun seudun ammattikorkeakoulu

TIIVISTELMÄ

Oulun seudun ammattikorkeakoulu
Tietotekniikan koulutusohjelma, ohjelmistojen kehitys

Tekijä: Pekka Lehtola

Opinnäytetyön nimi: Oulu UAS for Students - mobiilisovellus Android-alustalle

Työn ohjaajat: Pertti Heikkilä (ohjaava opettaja), Tommi Sallinen (tilaaja), Pekka Nisula (tilaajan edustaja)

Työn valmistumislukukausi ja -vuosi: Syksy 2012

Sivumäärä: 31 + 2

liitettä

Työn tarkoituksena oli toteuttaa Oulun seudun ammattikorkeakoulun Tekniikan yksikön tilauksesta opiskelijoiden käyttöön tarkoitettu Android-mobiilisovellus. Tavoitteena oli, että opiskelijat voisivat sovellusta käyttäen tarkastella kätevästi opiskelussa päivittäin tarvittavia tietoja kuten lukujärjestystä ja ruokalistaa omalla Android-älypuhelimellaan.

Tilaajan vaatimuksesta ohjelma rajattiin toimimaan laitteissa, joissa on alustana Android 2.2 (API 8) tai uudempi. Työ toteutettiin Java-kielellä käyttäen Googlen Android-alustalle laatimia Java-kirjastoja. Tarvittava tietoperusta on pääasiassa Android Developers -sivustolta. Teoria on peräisin suurimmaksi osaksi ajantasaisista verkkolähteistä.

Työn tuloksena saatiin tilaajan vaatimusten mukainen sovellus, jolla voi katsoa luokkakohtaista lukujärjestystä, ruokalistaa, Avack-pikatiedotteita sekä yksikön tiedotteita. Sovellus toimii sekä suomeksi, että englanniksi. Tilaajan puolella viime hetkellä tapahtuneiden tietokantamuutosten vuoksi sovelluksen julkaisu Google Playssa ei ollut mahdollista tämän työn puitteissa.

Asiasanat: Android, Java, mobiilisovellukset, opiskelijapalvelut, PHP

ABSTRACT

Oulu University of Applied Sciences
Information Technology, Software Development

Author: Pekka Lehtola

Title of thesis: UAS for Students -mobile application for Android

Supervisors: Pertti Heikkilä, Tommi Sallinen, Pekka Nisula

Term and year when the thesis was submitted: Autumn 2012 Pages: 31 + 2
appendices

The purpose of this thesis was to develop the Android mobile application for students of Oulu University of Applied Science, School of engineering. The aim was to view an convenient information about daily studying necessities such as a schedule and a lunch list presented by the application on screens of their Android mobile devices.

The customer requirement for the program was limited to Android devices with a platform revision of 2.2 (API 8) or newer. The work was implemented in Java included Java libraries for Android platform which developed by Google. Required information is mainly based on Android Developers site. The theory is derived mostly from up to date online sources.

As a result was an application that you can watch schedule by class, lunch list, news and Avack news. This application works in English or in Finnish.

Keywords: Android, Java, mobile software, PHP, student services

ALKULAUSE

Haluan kiittää Oulun seudun ammattikorkeakoulua ja suunnittelija Tommi Sallista, että sain tehdä tämän opiskelijoita tulevaisuudessa suuresti hyödyttävän työn. Haluan kiittää myös suunnittelija Pekka Nisulaa ja tuntiopettaja Pertti Heikkilää hyvästä ohjauksesta. Lisäksi haluan kiittää lehtori Tuula Hopeavuorta kielenohjauksesta.

6.9.2012

Pekka Lehtola

SISÄLLYS

TIIVISTELMÄ	3
ABSTRACT	4
ALKULAUSE	5
SISÄLLYS	6
SANASTO	8
1 JOHDANTO	10
2 VAATIMUSMÄÄRITTELY	11
2.1 Toiminnalliset vaatimukset	11
2.2 Tekniset vaatimukset	11
3 MENETELMÄT JA TYÖKALUT	12
3.1 Scrum-sovelluskehitysmalli	12
3.2 Työkalut	12
3.2.1 Eclipse	12
3.2.2 ADT	12
3.2.3 Android SDK	13
3.2.4 Notepad++	13
3.2.5 TortoiseSVN	13
4 ANDROID-MOBIILIOHJELMOINTI	14
4.1 Activityt ja XML-tyylitiedostot	14
4.2 Intentit	16
4.3 Eventit	17
4.4 AndroidManifest	18
5 TOTEUTUS	19
5.1 Projektin aloitus	19
5.2 Ensimmäinen sprintti	19
5.3 Toinen sprintti	20
5.4 Kolmas sprintti	21
5.5 Testaus	24
6 TULOKSET	25
7 YHTEENVETO	29
LÄHTEET	30

LIITE 1 AndroidManifest.xml

LIITE 2 main.xml

SANASTO

Android	Googlen omistama Linux-pohjainen avoimen lähdekoodin käyttöjärjestelmä mobiililaitteille.
API	Application programming interface eli ohjelmointirajapinta on määritelmä, jonka mukaan eri ohjelmat voivat tehdä pyyntöjä ja vaihtaa tietoja eli keskustella keskenään.
Java	Sun Microsystemsin kehittämä oliopohjainen ohjelmointikieli. Se on laitteistoriippumaton, läheisesti C++:aa muistuttava ja helposti omaksuttava.
JSON	JavaScript Object Notation on yksinkertainen tiedonsiirtomuoto, jota on helppo käyttää JavaScript-ohjelmissa. JSON on kuitenkin JavaScriptistä riippumaton, eli sitä voidaan käyttää myös muilla ohjelmointikielillä.
MySQL	MySQL on relaatiotietokantaohjelmisto. MySQL-tietokanta on hyvin suosittu web-palveluiden tietokantana. Sen päälle rakennettava ohjelmalogiikka tehdään usein PHP-, Python- tai Perl-ohjelmointikielellä. Sivut voidaan julkaista Apache-webpalvelimella, joka voi toimia Windows- tai Linux-käyttöjärjestelmän päällä.
PHP	PHP: Hypertext Preprocessor on Perlin kaltainen ohjelmointikieli, jota käytetään erityisesti Web-palvelinympäristöissä dynaamisten web-sivujen luonnissa.
RSS	Really Simple Syndication on joukko verkkosyötemuotoja, joita käytetään usein päivittyvän digitaalisen sisällön julkaisemiseen. Tällaista sisältöä ovat esimerkiksi blogit, uutiset ja podcastit.

XML

Extensible Markup Language on merkintäkieli tai standardi, jolla tiedon merkitys on kuvattavissa tiedon sekaan. Sitä käytetään sekä formaattina tiedonvälitykseen järjestelmien välillä että formaattina dokumenttien tallentamiseen. Se on rakenteellinen kuvauskieli, joka auttaa jäsentämään laajoja tietomassoja selkeämmin.

1 JOHDANTO

Työn tarkoituksena oli kehittää OAMK:n (Oulun seudun ammattikorkeakoulun) Tekniikan yksikölle opiskelijoiden käyttöön tarkoitettu mobiilisovellus Android-alustalle. Sovelluksen on tarkoitus helpottaa opiskelijoiden elämää mahdollistamalla lukujärjestyksen, tiedotteiden, ruokalistan jne. tarkastelu suoraan omasta älypuhelimesta tai muusta mobiililaitteesta. Aiemmin opiskelijoiden on pitänyt tarkastella edellä mainittuja tietoja eri lähteistä, eikä se välttämättä ole ollut olleenkaan mahdollista mobiililaitteilla.

Sovellusta oli kehitetty jo aiemmin harjoitteluprojektina ja niinpä pohjaksi tähän työhön otettiin harjoitteluprojektissa tehty suppea ja hiomattomaksi jäänyt sovellus. Tässä työssä keskityttiin siten sovelluksen sisällön laajentamiseen ja käytettävyyden parantamiseen. Sovelluksessa näytettävät sisällöt haetaan pääasiassa OAMK:n Tekniikan yksikön palvelimelta, mutta lisäksi jonkin verran myös internetistä. Tarkoituksena oli, että sovellus julkaistaisiin Google Play -Android-sovelluskaupassa.

2 VAATIMUSMÄÄRITTELY

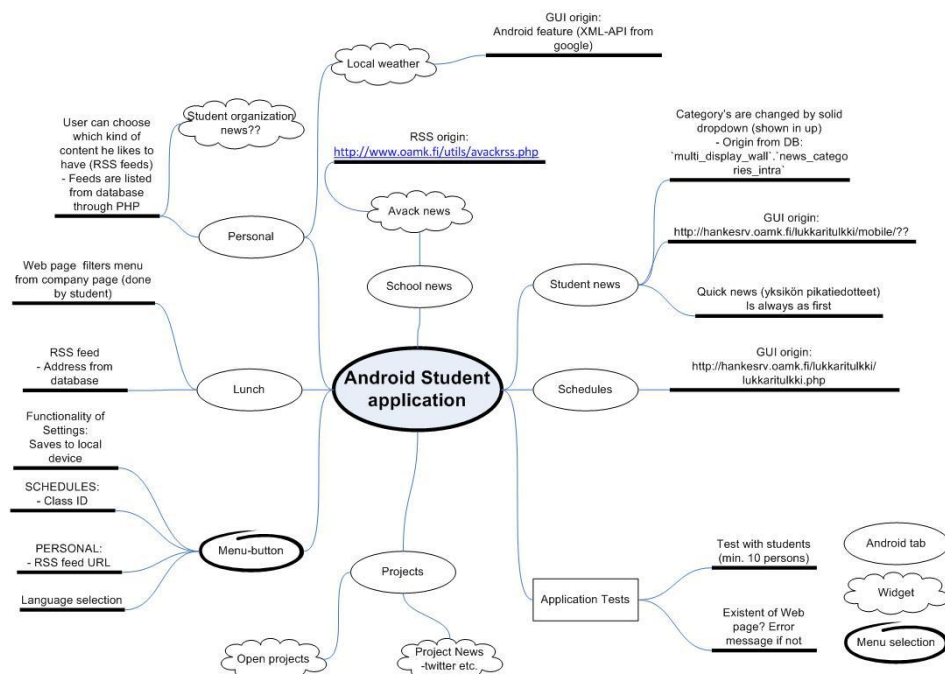
2.1 Toiminnalliset vaatimukset

Päävaatimukset olivat, että sovelluksen kautta voi tarkastella lukujärjestystä, ruokalistaa, tiedotteita ja Avack-pikatiedotteita. Sovelluksen piti lisäksi olla käytävissä sekä suomeksi että englanniksi.

2.2 Tekniset vaatimukset

Sovelluksen piti toimia Android API 8:ssa ja uudemmissa versioissa. Ulkoasun piti olla väritykseltään yhtenäinen OAMK:n muiden sähköisten palvelujen ilmeen kanssa. Toimintojen piti olla käyttäjän kannalta ajateltuna yksinkertaisia ja loogisia.

Tilaa antoi vaatimusmäärittelyn käsitekarttana (kuva 1). Käsitekartassa mainitut Personal- ja Projects-osiot sekä testaus opiskelijaryhmällä jäivät mahdolliseen jatkokehitysvaiheeseen.



KUVA 1. Käsitekartta, joka sisältää toiminnalliset ja osin myös tekniset vaatimukset

3 MENETELMÄT JA TYÖKALUT

3.1 Scrum-sovelluskehitysmalli

Scrum tarjoaa sovelluskehitykseen mallin, jonka mukaan projektia ohjataan. Scrum ei ota kantaa matalan tason insinöörikäytäntöihin, vaan keskittyy ennen muuta projektin vaiheistamiseen ja jatkuvaan kontrolliin projektin etenemisestä. Näin Scrum soveltuu etenkin asiakkaille hyväksi johdannoksi ketterien menetelmien maailmaan. (1.)

Scrum - kuten kaikki muutkin ketterät mallit - näkee ohjelmistokehityksen rakentuvan erimittaisten syklien ympärille. Tärkeimmät syklit ovat sprintti ja päivä. Sprintillä tarkoitetaan yhtä kehitysjaksoa, jonka jälkeen tuote on ainakin periaatteessa julkaisuvalmis. Tyypillisesti sprintin kesto on kuukausi, mutta sen pituus vaihtelee organisaation tarpeiden mukaan viikosta kahteen kuukauteen. (1.)

Sprintin aikana tiimi toteuttaa sprinttiin kuuluviksi valittuja toiminnallisuuksia. Sprintin aikana vaatimusten muuttaminen on kiellettyä, ja tiimillä on täysi vapaus tehdä tarpeelliseksi katsomiaan toimenpiteitä, jotta sovittu sprintin päämäärä voidaan saavuttaa. Tiimi organisoi itsensä parhaaksi katsomallaan tavalla. (1.)

3.2 Työkalut

3.2.1 Eclipse

Eclipse on ohjelmointiympäristö, joka tukee mm. Java-, C-, C++- ja PHP-kieliä. Eclipsen on alun perin kehittänyt IBM, mutta nykyään sen kehityksestä vastaa Eclipse Foundation. Eclipsen lähdekoodi on ollut avoin vuodesta 2001 lähtien. (3.)

3.2.2 ADT

ADT (Android Development Tools) on Eclipsen liitännäinen, joka integroi Android-kehitystyökalut Eclipseen.

3.2.3 Android SDK

Android Software Development Kit (SDK) on paketti, joka sisältää kattavan valikoiman kehitystyökaluja, mm. virheenkorjaustyökalun (debuggerin), kirjastot ja puhelinemulaattorin.

3.2.4 Notepad++

Notepad++ on avoimeen lähdekoodiin pohjautuva tekstieditori Windowsille. Se tukee useiden eri ohjelmointikielien syntaksin korostusta, useiden tekstien yhtäaikaista avaamista välilehtiin sekä hakua ja korvausta avoimena olevista tiedostoista myös säännöllisillä lausekkeilla.

3.2.5 TortoiseSVN

TortoiseSVN on ilmainen, avoimeen lähdekoodiin perustuva Windows-käyttöliittymä Apache™ Subversion®-versionhallintaan. Se pitää kirjaa tiedostoihin ja hakemistoihin ajan mittaan tehdyistä muutoksista. Tiedostot talletetaan keskitettyyn arkistoon. (7.)

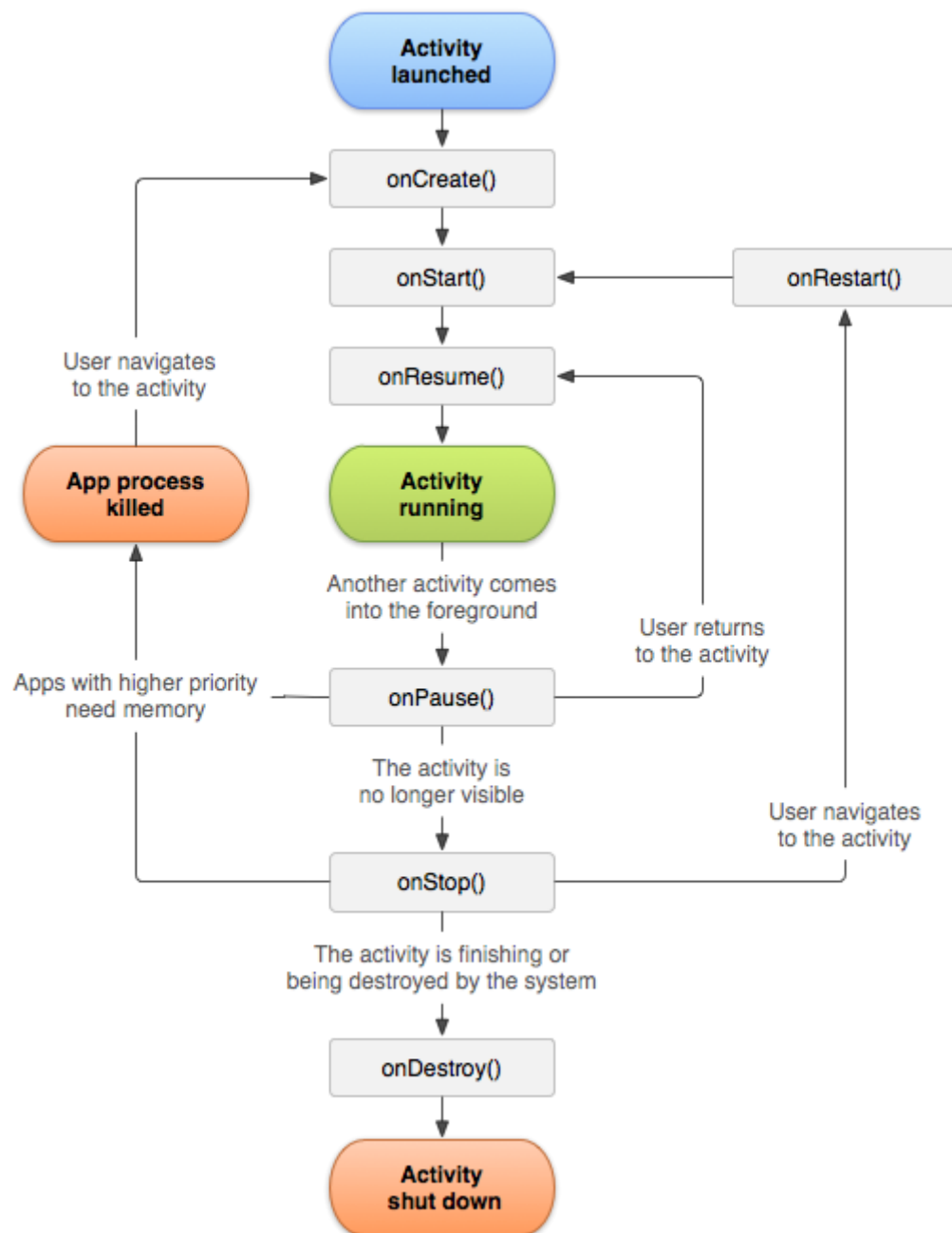
4 ANDROID-MOBIILIOHJELMOINTI

Android-sovellukset toimivat Javaan perustuvan Dalvik-virtuaalikoneen päällä. Android-sovelluskehitys tehdään Java-kielellä ja Google tarjoaa SDK:n ilmaiseksi. Google tarjoaa myös Eclipse-pluginin, joka tarjoaa automaattisen sovelluksen paketoinnin apk-tiedostoksi ja sovelluksen ajamisen ja testaamisen Android-emulaattorissa. (8.)

Tarkalleen ottaen kehitystä ei tehdä virallisella Java-kielellä, sillä Google käyttää Apache Harmony -luokkakirjastoja ja niistä luotu Java-tavukoodi käännetään vielä erikseen Dalvikin käyttämään muotoon. Harmony tarjoaa lähestulkoon samat luokkakirjastot kuin standardi-Java. (8.)

4.1 Activityt ja XML-tyylitiedostot

Android-mobiiliohjelmoinnissa sovelluksen toiminnallisuus ja ulkoasu erotetaan yleensä tarkasti toisistaan. Toiminnallisuus toteutetaan activityissa (kuva 2) ja ulkoasu XML (Extensible Markup Language) -tyylitiedostoissa.



KUVA 2. Activityn toiminta kaaviona esitettynä (9)

Tässä kyseisessä työssä on kahdeksan activityä. Jokainen välilehti on oma activitynsä. Lisäksi on mainactivity, johon välilehdet on upotettu, sekä kielen-, koulutusohjelman- ja luokanvalinta-activityt. Jokainen activity käyttää samaa main.xml-tyylitiedostoa, jossa perusulkoasu on määritelty (liite 2), eli jokaisella activityllä on sama layout. Lisäksi jokaisella välilehdellä oma XML-tyylitiedostonsa, joissa määritellään välilehtien valintakuvakkeet.

XML-tyylitiedostojen avulla voidaan toteuttaa tavallaan myös toiminnallisuutta. Tässä työssä toteutettiin välilehtien valintakuvakkeiden vieritys lisäämällä tarvittava koodi mainactivityn XML-tyylitiedostoon (5). Toiminto tosin näkyy käytännössä vasta sitten, jos jatkokehityksen myötä välilehtien määrää lisätään.

Aina ulkoasua ei kuitenkaan voi toteuttaa pelkästään XML-tyylitiedostoja käyttäen. Tiedote-välilehdellä tarvittiin ulkoasua, jollaista ei muualla sovelluksessa tarvittu. Niinpä se toteutettiin koodaamalla ulkoasu activityn koodiin. Tehdyssä toteutuksessa mukailtiin internetistä löytynyttä esimerkkiä (6).

4.2 Intentit

Intent on luokka, joka sisältää toiminnon abstraktin kuvauksen, eli käytännössä lähetetään kutsuvan aktiviteetin tiedot kutsuttavalle toiselle aktiviteetille. Intentin rooli on toimia sitovana elementtinä sovelluksessa, jossa on useita eri aktiviteetteja. Sen välityksellä on mahdollista siirtää arvoja kahden eri aktiviteetin välillä (koodi 2). (10.)

Oulu UAS for Students -sovelluksessa intentiä käytetään, kun välitetään koulutusohjelman lyhenne SettingsActivitystä ClassActivityyn (koodi 1, 2 ja 3). Kun valitaan koulutusohjelma (kuva 3), välitetään koulutusohjelman lyhenne, jonka perusteella ClassActivity hakee mahdolliset luokkatunnukset valikkoon (kuva 4).

```
57      intent = new Intent(this, ClassActivity.class);
```

KOODI. 1. Activityn määrittely intentiin

```
99      intent.putExtra("lyhenne", lyhenne );  
100     startActivity(intent);
```

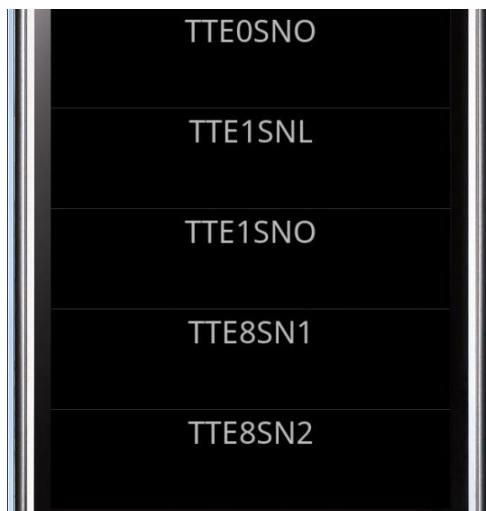
KOODI. 2. Koulutusohjelman lyhenteen siirto ClassActivityyn intentiä käyttäen

```
54      lyhenne = this.getIntent().getExtras().getString("lyhenne");
```

KOODI. 3. Koulutusohjelman lyhenteen vastaanotto ClassActivityssä



KUVA 3. Koulutusohjelmanvalinta



KUVA 4. Luokanvalinta

4.3 Eventit

Jokainen activity sisältää view'n ja esimerkiksi jokainen view:ssä tapahtuva napautus tai napin painallus on eventti eli tapahtuma. Tapahtuman käsittelyyn tarvitaan sopiva Listener. Tässä sovelluksessa esimerkiksi koulutusohjelman valinta SettingsActivityssä napauttamalla valikkolistasta on tapahtuma. Kun Listener huomaa napautuksen, se kutsuu onClick-metodia, jossa valitun koulutusohjelman lyhenne otetaan talteen muuttujaan ja toimitetaan intentiä käyttäen ClassActivitylle (koodi 4).

```

90
91
92 tv.setOnClickListener(new OnClickListener() {
93     @Override
94     public void onClick(View arg0) {
95         //Toast.makeText(getApplicationContext(), ((TextView) tv).getText(),
96         //    Toast.LENGTH_SHORT).show();
97         System.out.println(lyhenteet.get(position));
98         // TODO Auto-generated method stub
99         System.out.println(tv.getText());
100         lyhenne = lyhenteet.get(position).toString();
101         intent.putExtra("lyhenne", lyhenne );
102         startActivity(intent);
103         finish();
104     }
});

```

KOODI. 4. Eventin eli tapahtuman käsittely

4.4 AndroidManifest

AndroidManifest on XML-tiedosto, jossa määritellään kaikki sovelluksessa suoritettavat activityt. Jos activityä ei ole määritelty AndroidManifestissä, se jää konaan suorittamatta. AndroidManifestissä myös määritellään se, mikä activity suoritetaan ensimmäisenä, kun sovellus käynnistetään. Jos sovelluksen pitää voida ottaa yhteys internetiin, AndroidManifestissä pitää antaa sovellukselle siihen oikeus. (Liite 1.)

5 TOTEUTUS

5.1 Projektin aloitus

Aloituspalaverissa sovittiin tilaajan ja ohjaajan kanssa vaatimuksista ja alustavasta aikataulusta. Sen jälkeen laadittiin suunnitelma, joka sisälsi tarkemman aikataulun sekä työvaiheet ja niiden jaottelun sprintteihin. Suunnitelma piti hyväksyttää ohjaavalla opettajalla.

5.2 Ensimmäinen sprintti

Ensimmäisessä sprintissä oli tarkoitus viimeistellä harjoitteluprojektissa tehdyn sovelluksen pohjalta lukujärjestys- ja ruokalistavälilehdet. Ruokalistavälilehden toimivuutta ajateltiin parantaa hakemalla ruokalista ravintolan sivuilta RSS (Really Simple Syndication) -syötettä käyttäen. RSS-syötteen ajateltiin olevan kätevä välilehden päivitykseen. Toteutuksen piti tapahtua siten, että internetistä etsitään valmis RSS-lukija, jota olisi muokattu tarpeen mukaan ja yhdistetty siten sovellukseen.

Netistä haetun RSS-lukija sovelluksen kanssa tuli kuitenkin ongelmia hyvin pian. RSS-syötteet, joita oli tarkoitus käyttää, eivät näkyneet oikein. Merkillisintä oli se, että vaikka RSS-syötteet oli koodattu kaikki samalla lailla standardin mukaisesti, ei ongelma toistunut jokaisessa RSS-syötteessä samanlaisena. Osa RSS-syötteistä näkyi osittain, osa ei ollenkaan. Yksikään RSS-syöte ei näkynyt oikein. Aikataulun tiukkuuden ja vähäisen kokemuksen takia RSS-syötteisiin ratkaisua ei ehditty löytää, joten ruokalistavälilehti päätettiin toteuttaa PHP (PHP: Hypertext Preprocessor) -tiedostoa käyttäen.

PHP-tiedostolla haetaan ensin ruokalistan sisältö ravintolan verkkosivuilta. Koska ruokalistan osoite sisältää päivämäärän ja näin ollen muuttuu joka päivä, päivitetään PHP-tiedostossa osoitteeseen oikea päivämäärä. Sen jälkeen tarpeelliset tiedot parseroidaan haetusta sisällöstä ja tulostetaan mobiililaitteen ruudulle sopivassa muodossa webview-näyttöelementtiin (2). PHP-tiedostoja laadittiin kaksi, joista toinen hakee suomen- ja toinen englanninkielisen ruokalistan. Ruokalista tulostetaan ruudulle aina kielivalinnan mukaisesti.

5.3 Toinen sprintti

Toisessa sprintissä oli alun perin tarkoitus toteuttaa Avack-pikatiedote- ja tiedotevälilehdet. Koska ruokalistavälilehden toteutus RSS-syötteellä ei onnistunut, jouduttiin toisesta sprintistä kuitenkin osa käyttämään ruokalistavälilehden toteuttamiseen PHP-tiedostoilla. Myös Avack-pikatiedotevälilehti toteutettiin PHP-tiedostoa käyttämällä. Ruokalista- ja Avack-pikatiedotevälilehdillä käytettiin kuten lukujärjestys- ja myöhemmin tiedotevälilehdelläkin AsyncTaskia. Sen avulla saatiin luotua dialogi, jota näytetään välilehdillä aina siihen asti, että sisältö saadaan ladattua (koodi 5, koodi 6).

```
60     dialog = new ProgressDialog(this);
61     dialog.setTitle("Sivua ladataan/Loading page");
62     dialog.setMessage("Odota.../Wait...");
63     dialog.show();
64
65     webpage = "http://hankesrv.oamk.fi/lukkaritulkki/mobile//avack.php";
66
67     try {
68         new MyTask().execute(webpage);
69     } catch (Exception e) {
70         // TODO Auto-generated catch block
71         e.printStackTrace();
72     }
73
```

KOODI 5. Avack-pikatiedotevälilehden koodi, jossa luodaan dialogi, asetetaan PHP-tiedoston osoite muuttujaan ja kutsutaan AsyncTaskia

```

76  class MyTask extends AsyncTask<String, Integer, Boolean> {
77      protected void onPreExecute(Long result) {
78          ...
79      }
80
81      @Override
82      protected Boolean doInBackground(String... params) {
83          // TODO Auto-generated method stub
84          webview.getSettings().setBuiltInZoomControls(true);
85          webview.setInitialScale(50);
86          webview.loadUrl(webpage);
87          return true;
88      }
89      protected void onPostExecute(Boolean result) {
90          ...
91          setContentView(webview);
92          dialog.dismiss();
93      }
94
95
96
97
98
99  }

```

KOODI 6. *MyTask-niminen AsyncTask*

5.4 Kolmas sprintti

Kolmannessa sprintissä toteutettiin tiedotevälilehti. Tiedotevälilehti toteutettiin siten, että välilehdelle laitettiin pudotusvalikko, josta valitaan haluttu tiedote-kategoria (koodi 7). Tähän löytyi malli Androidin Developers-sivulta (4).

```

141  spinner = new Spinner(this);
142
143  layout.addView(spinner, labelLayoutParams);
144
145  //ArrayAdapter<String> adapter = new ArrayAdapter<NewsActivity.this, android.R.layout.simple_spinner_item, id>;
146
147  adapter = new ArrayAdapter<NewsActivity.this, android.R.layout.simple_spinner_item, categoria>;
148
149  spinner.setAdapter(adapter);
150
151  spinner.setOnItemSelectedListener(new MyOnItemSelectedListener());
152

```

KOODI 7. *Koodi, jolla luodaan pudotusvalikko (spinner)*

Kategorioiden nimet haetaan pudotusvalikkoon MySQL-tietokannasta JSONia (JavaScript Object Notation) käyttäen (koodi 8, 9 ja 10).

```

173     JSONObject json_data;
174     JSONArray jArray;
175     String result = "";
176     //the year data to send
177     ArrayList<NameValuePair> nameValuePairs = new ArrayList<NameValuePair>();
178     nameValuePairs.add(new BasicNameValuePair("year","1980"));
179     InputStream is = null;
180     //http post
181     try{
182         HttpClient httpclient = new DefaultHttpClient();
183         HttpPost httppost = new HttpPost("http://hanke.srv.oamk.fi/lukkaritulkki/mobile/news_id.php");
184         //HttpPost httppost = new HttpPost("http://icdubilab1.oamk.fi/t8lepe00/getprograms.php");
185         //HttpPost httppost2 = new HttpPost("http://www.oamk.fi/~peninsula/t8lepe00/getClasses.php?class=AUT");
186         httppost.setEntity(new UrlEncodedFormEntity(nameValuePairs));
187         HttpResponse response = httpclient.execute(httppost);
188         HttpEntity entity = response.getEntity();
189         is = entity.getContent();
190         Toast.makeText(this,"Done", Toast.LENGTH_LONG).show();
191     }catch(Exception e){
192         Log.e("log_tag", "Error in http connection "+e.toString());
193         AlertDialog.Builder builder = new AlertDialog.Builder(this);
194         builder.setMessage("Yhteyttä ei ole! / No connection!")
195             .setCancelable(false)
196             .setPositiveButton("Ok", new DialogInterface.OnClickListener() {
197                 public void onClick(DialogInterface dialog, int id) {
198                     //dialog.cancel();
199                     NewsActivity.this.finish();
200                 }
201             });
202         AlertDialog alert = builder.create();
203         alert.show();
204     }
205 }

```

KOODI 8. Koodi, jolla muodostetaan yhteys PHP-tiedostoon, joka hakee tiedot MySQL-tietokannasta

```

206     //convert response to string
207     try{
208         BufferedReader reader = new BufferedReader(new InputStreamReader(is,"iso-8859-1"),8);
209         StringBuilder sb = new StringBuilder();
210         String line = null;
211         while ((line = reader.readLine()) != null) {
212             sb.append(line + "\n");
213         }
214         is.close();
215         result=sb.toString();
216     }catch(Exception e){
217         Log.e("log_tag", "Error converting result "+e.toString());
218     }
219 }
220

```

KOODI 9. Koodi, joka sijoittaa haetut tiedot result-muuttujaan

```

223 jsonArray = new JSONArray(result);
224 for(int i=0;i<jArray.length();i++){
225     json_data = jsonArray.getJSONObject(i);
226     if (json_data.getString("nimi") != "null")
227     {
228         if (kieli == "Suomi")
229         {
230             id_list.add(json_data.getString("id"));
231             categoria.add(json_data.getString("nimi"));
232         }
233         else if (kieli == "English")
234         {
235             id_list.add(json_data.getString("id"));
236             categoria.add(json_data.getString("nimi_en"));
237         }
238         else
239         {
240             id_list.add(json_data.getString("id"));
241             categoria.add(json_data.getString("nimi_en"));
242         }
243     }

```

KOODI 10. Koodi, joka parseroi MySQL-tietokannasta haetut, result-muuttujaan sijoitetut kategoria tiedot kielen mukaan

Kun kategoria on valittu, haetaan kategoriassa olevat tiedotteet tietokannasta erillistä verkkosovellusta käyttäen. Tämä tapahtuu siten, että kun kategoria on valittu, haetaan kategorian indeksi taulukosta. Kun indeksi on haettu, se sijoitetaan verkkosovelluksen nettiosoitteen loppuun. Indeksiä vastaavat tiedotteet näkyvät webview:ssä, joka on toteutettu Androidin Developers-sivulta löytyvän mallin mukaan (2). Kielivalinta määrää kummalla kielellä tiedotteet näkyvät. Jos tiedotetta ei ole englanniksi, se näytetään suomeksi. (KOODI 11.)

```

274
275 /*Toast.makeText(parent.getContext(), "The planet is " +
276     testi, Toast.LENGTH_LONG).show();*/
277
278 if (kieli == "Suomi")
279 {
280     webpage = "http://hankesrv.oamk.fi/lukkaritulikki/mobile/JS_student_news/index.php/news/show_news/" + selected_id;
281     webview.loadUrl(webpage);
282 }
283 else if (kieli == "English")
284 {
285     webpage = "http://hankesrv.oamk.fi/lukkaritulikki/mobile/JS_student_news_en/index.php/news/show_news/" + selected_id;
286     webview.loadUrl(webpage);
287 }
288 else
289 {
290     webpage = "http://hankesrv.oamk.fi/lukkaritulikki/mobile/JS_student_news_en/index.php/news/show_news/" + selected_id;
291     webview.loadUrl(webpage);
292 }
293
294

```

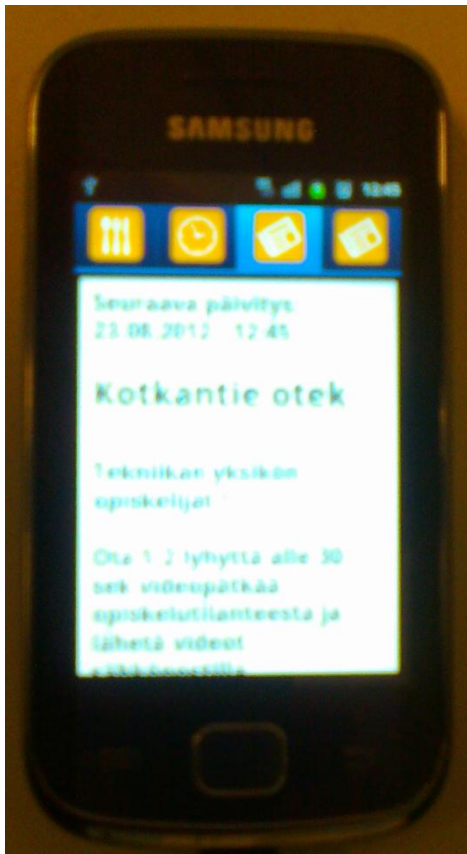
KOODI 11. Koodi, joka hakee tiedotteet kielivalinnan määräämällä kielellä valitun kategorian indeksin mukaan

Tiedotteet voi halutessaan avata napauttamalla, mikäli tiedotteessa on muuta-
kin sisältöä kuin otsikko. Tiedotteen sulkeminen tapahtuu myös napauttamalla.

5.5 Testaus

Testaus tapahtui Eclipse-ympäristössä emulaattoria käyttäen. Testausta tehtiin
projektin edetessä koko ajan. Kun lisättiin uusia ominaisuuksia tai muokattiin
olemassa olevia ominaisuuksia, ne testattiin saman tien. Näin virheet voitiin
huomata nopeasti, jolloin ne voitiin myös korjata pian.

Sovellus testattiin lopuksi myös Android-älypuhelimessa (kuva 5). Näin saatiin
varmistettua, että sovellus toimii myös uusimmalla Android 4.0 (API 15)
-alustalla.



KUVA 5. Sovellus testattavana Samsung-merkkisessä Android-älypuhelimessa

6 TULOKSET

Työn tuloksena saatiin toimiva sovellus, jolla voi kätevästi tarkastella ruokalistaa, lukujärjestyksiä, Avack-pikatiedotteita ja tiedotteita. Sovelluksessa on neljä välilehteä, joista kukin sisältää yhden edellä mainituista toiminnoista. Lisäksi on kielivalikko ja asetusvalikko, josta voi valita koulutusohjelman ja luokan lukujärjestystä varten.

Kun sovelluksen käynnistää ensimmäisen kerran, on oletuskielenä englanti. Kielen voi vaihtaa halutessaan suomeksi. Kieli valitaan painamalla ensin puhelimen menu-painiketta. Sen jälkeen painetaan kielestä riippuen Kieli- tai Language-painiketta, jolloin avautuu lista, josta voidaan valita haluttu kieli. Vaihtoehtoina ovat suomi ja englanti.

Lukujärjestysvälilehdelle mentäessä ensimmäisen kerran näytetään ohjeet koulutusohjelman- ja luokanvalintaan. Koulutusohjelma ja luokka valitaan siten, että ensin painetaan puhelimen menu-painiketta. Sen jälkeen painetaan kielestä riippuen joko Asetukset- tai Settings-painiketta, jolloin avautuu lista, josta voidaan valita haluttu koulutusohjelma. Kun koulutusohjelma on valittu, avautuu lista, josta voidaan valita haluttu luokka. Tämän jälkeen ohjelma menee Avack-pikatiedotevälilehteen, josta voidaan palata lukujärjestysvälilehteen, jolloin nähdään koulutusohjelma- ja luokkavalinnan mukainen lukujärjestys (kuva 6).




Uutis 2011 Teknillisen korkeakoulun kampus OULUN SEUDUN ASMAATTI 27.04.12
18.53

TTEISNO Tietotekniikka, Ohjelmistojen Jyrkä Kari

OHJELMA	OHJELMA	OHJELMA	OHJELMA
6:15	Ohjelmaohjelma, Ohjelmaohjelma		
9:00	Ohjelmaohjelma, Ohjelmaohjelma		
9:15	Ohjelmaohjelma, Ohjelmaohjelma		
10:00	Ohjelmaohjelma, Ohjelmaohjelma		
10:15	Ohjelmaohjelma, Ohjelmaohjelma		
11:00	Ohjelmaohjelma, Ohjelmaohjelma		
11:15	Ohjelmaohjelma, Ohjelmaohjelma		
12:00	Ohjelmaohjelma, Ohjelmaohjelma		
12:15	Ohjelmaohjelma, Ohjelmaohjelma		
13:00	Ohjelmaohjelma, Ohjelmaohjelma		
13:15	Ohjelmaohjelma, Ohjelmaohjelma		
14:00	Ohjelmaohjelma, Ohjelmaohjelma		
14:15	Ohjelmaohjelma, Ohjelmaohjelma		
15:00	Ohjelmaohjelma, Ohjelmaohjelma		
15:15	Ohjelmaohjelma, Ohjelmaohjelma		
16:00	Ohjelmaohjelma, Ohjelmaohjelma		
16:15	Ohjelmaohjelma, Ohjelmaohjelma		
17:00	Ohjelmaohjelma, Ohjelmaohjelma		
17:05	Ohjelmaohjelma, Ohjelmaohjelma		
17:50	Ohjelmaohjelma, Ohjelmaohjelma		
18:05	Ohjelmaohjelma, Ohjelmaohjelma		
18:50	Ohjelmaohjelma, Ohjelmaohjelma		
19:00	Ohjelmaohjelma, Ohjelmaohjelma		
19:45	Ohjelmaohjelma, Ohjelmaohjelma		
19:45	Ohjelmaohjelma, Ohjelmaohjelma		
20:30	Ohjelmaohjelma, Ohjelmaohjelma		

KUVA 6. Lukujärjestysnäköymä

Kun käyttäjä valitsee ruokalistavälilehden, näytetään viikon ruokalista, jos se on saatavilla (kuva 7).



Lounas 11.06.-17.06.

Maanantai
Porsaanlihaströgonoffi VI.
Chili on carne G, M
Lounassaati G

Tiistai
Lihapullat ja peruna VI.
Kavio-pastavoka L
Cheddarjuustosaati G, VI

Keskiviikko
Talon kulapohut, olikasike ja keitetty peruna VI.
Kavio-pastavoka M
Lounassaati G, M

Torstai
Munnapuuro ja mehuaiketta, jalkiruoka.
Kalkkuna Cordon bleu ja risi VI.
Lounassaati

Perjantai
Kinkkukiusaus G, VI.
Kavio-pastavoka G, VI.
Lounassaati G, VI.

VI = vatsatauti / low lactose / allergia L = laktoosi / lactose free / laktoosi M = maidoton / milk free / ei-maitoa G = gluteniton / gluten free / gluteniton Ravintolamme henkilökunta auttaa mieleniäsi ruoka-
allergioihin liittyvissä asioissa Our restaurant personnel will be happy to help you concerning special diets or
allergies. Muutokset ruokalistalla ovat mahdollisia. Variation is possible within the menu.

KUVA 7. Ruokalistanäkymä

Kun käyttäjä valitsee Avack-pikatiedotevälilehden, näytetään uusimmat pikatie-
dotteet (kuva 8).



KUVA 8. Avack-pikatiedotenäkymä

Kun käyttäjä valitsee tiedotevälilehden, ruudulle tulee pudotusvalikko, josta käyttäjä voi valita haluamansa tiedote kategorian. Kun kategoria on valittu, näytetään kyseisessä kategoriassa olevat tiedotteet webview:ssä. Oletuksena on valittuna Tiedoksi-kategoria, jonka tiedotteet näkyvät valmiina webview:ssä (kuva 9).



KUVA 9. Tiedotenäkymä

7 YHTEENVETO

Tämän työn tarkoituksena oli kehittää OAMK:n Tekniikan yksikön opiskelijoiden käyttöön opiskelussa päivittäin tarvittavien tietojen hakua helpottava sovellus. Sovelluksella oli tarkoitus pystyä katsomaan lukujärjestystä, ravintolan ruokalista, Avack-pikatiedotteita ja yksikön tiedotteita. Sovelluksen piti toimia Android-älypuhelimissa ja -tableteissa. Työn tuloksena syntyi tilaajan vaatimukset täyttävä sovellus. Ajan puutteen ja tilaajan puolella viime hetkellä tapahtuneiden tietokantamuutosten vuoksi sovelluksen julkaiseminen ei ole vielä mahdollista, joten julkaisu Google Playssa jäi pois tästä opinnäytetyöstä.

Työstä oli minulle paljon hyötyä. Sain paljon kokemusta Android-ohjelmistokehityksestä ja Eclipse-kehitysympäristöstä. Uskon, että voin hyödyntää saamaani kokemusta myöhemmin työelämässä hyvinkin paljon.

Mahdollisessa jatkekehityksessä voitaisiin ratkaista edellä mainitusta tietokantamuutoksesta johtuneet ongelmat. Lisäksi sovelluksen voisi optimoida toimimaan myös tablet-tietokoneissa ja ruokalista- ja Avac-pikatiedotevälilehtien ulkoasua voisi vielä parantaa.

LÄHTEET

1. Tolvanen, Perttu. 2011. Ketteryys haltuun: SCRUM pähkinänkuoressa. Saatavissa: <http://www.meteoriitti.com/fi-FI/tiedotteet/ajankohtaista/ketteryys-haltuun-scrum-pahkinankuoressa>. Hakupäivä: 8.8.2012.
2. WebView. 2012. Saatavissa: <http://developer.android.com/reference/android/webkit/WebView.html>. Hakupäivä: 8.8.2012.
3. Eclipse (IDE). 2012. Saatavissa: http://fi.wikipedia.org/wiki/Eclipse_%28IDE%29. Hakupäivä: 13.8.2012
4. Spinners. 2012. Saatavissa: <http://developer.android.com/guide/topics/ui/controls/spinner.html>. Hakupäivä: 8.8.2012.
5. Dyer, Dan. 2012. Scrolling Tabs in Android. Saatavissa: <http://blog.uncommons.org/2011/04/18/scrolling-tabs-in-android/>. Hakupäivä: 8.8.2012
6. How to set RelativeLayout layout params in code not in xml. 2011. Saatavissa: <http://stackoverflow.com/questions/5191099/how-to-set-relativelayout-layout-params-in-code-not-in-xml>. Hakupäivä: 8.8.2012.
7. Mikä TortoiseSVN on? 2012. Saatavissa: http://tortoisesvn.net/docs/release/TortoiseSVN_fi/tsvn-preface.html#tsvn-preface-about. Hakupäivä: 8.8.2012
8. Mikä on Android. 2012. Saatavissa: <http://blog.androidsuomi.fi/mika-on-android/>. Hakupäivä: 8.8.2012
9. Activity. 2012. Saatavissa: <http://developer.android.com/reference/android/app/Activity.html>. Hakupäivä: 13.8.2012

10. Android ohjelmointi - Tunti 5. 2011. Saatavissa:

http://some.lappia.fi/wiki/images/d/dc/Android_ohjelmointi_05.pdf. Hakupäivä: 23.8.2012

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      package="com.hellotabwidget"
4      android:versionCode="1"
5      android:versionName="1.0">
6      <uses-sdk android:minSdkVersion="8" />
7      <uses-permission android:name="android.permission.INTERNET" />
8
9      <application android:icon="@drawable/icon4" android:label="@string/app_name">
10         <activity android:name=".MobileStudentsApplicationActivity"
11             android:label="@string/app_name"
12             android:theme="@android:style/Theme.NoTitleBar">
13             <intent-filter>
14                 <action android:name="android.intent.action.MAIN" />
15                 <category android:name="android.intent.category.LAUNCHER" />
16             </intent-filter>
17         </activity>
18         <activity android:name=".MenuActivity"
19             android:label="@string/app_name"
20             android:theme="@android:style/Theme.NoTitleBar">
21         </activity>
22         <activity android:name=".ScheduleActivity"
23             android:label="@string/app_name"
24             android:theme="@android:style/Theme.NoTitleBar">
25         </activity>
26         <activity android:name=".AvackActivity"
27             android:label="@string/app_name"
28             android:theme="@android:style/Theme.NoTitleBar">
29         </activity>
30         <activity android:name=".NewsActivity"
31             android:label="@string/app_name"
32             android:theme="@android:style/Theme.NoTitleBar">
33         </activity>
34         <activity android:name=".SettingsActivity"
35             android:label="@string/app_name"
36             android:theme="@android:style/Theme.NoTitleBar">
37         </activity>
38         <activity android:name=".ClassActivity"
39             android:label="@string/app_name">
40         </activity>
41         <activity android:name=".LanguageActivity"
42             android:label="@string/app_name">
43         </activity>
44     </application>
45 </manifest>

```



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <TabHost xmlns:android="http://schemas.android.com/apk/res/android"
3     android:id="@android:id/tabhost"
4     android:layout_width="fill_parent"
5     android:layout_height="fill_parent">
6     <LinearLayout
7         android:orientation="vertical"
8         android:layout_width="fill_parent"
9         android:layout_height="fill_parent"
10        android:padding="5dp">
11        <HorizontalScrollView android:layout_width="fill_parent"
12            android:layout_height="wrap_content"
13            android:fillViewport="true"
14            android:scrollbars="none">
15            <TabWidget
16                android:id="@android:id/tabs"
17                android:layout_width="fill_parent"
18                android:layout_height="wrap_content" />
19        </HorizontalScrollView>
20        <FrameLayout
21            android:id="@android:id/tabcontent"
22            android:layout_width="fill_parent"
23            android:layout_height="fill_parent"
24            android:padding="5dp">
25            <MediaController android:id="@+id/mediaController1" android:layout_width="wrap_content" android:layout_height="wrap_content"></MediaController>
26        </FrameLayout>
27    </LinearLayout>
28 </TabHost>
```